

622, database 658 may reside in any suitable location or locations accessible by processor 650. Database 658 may include any hardware, software, firmware, or combination thereof suitable to store and facilitate retrieval of information. Database 658 may store and processor 650 may process any suitable information to perform  
5 order fulfillment operations in system 10. The following examples are for illustration only. Any other and/or additional types of information may be used without departing from the scope of the present invention.

In one embodiment, database 658 stores one or more product lists 660. Product lists 660 identify one or more products for which LFM 622 may generate  
10 component quotations 34 and component promises 44. For example, product list 660 may identify the name and/or code that identifies each product that a supplier associated with LFM 622 wants to sell. When LFM 622 receives a component ATP request 32 from fulfillment server 16, LFM 622 may examine product list 660 and determine if the product requested in the component ATP request 32 is supported by  
15 LFM 622. If so, LFM 622 may generate a component quotation 34 for that request 32.

Database 658 may also store one or more supply vectors 662. Supply vector 662 identifies when one or more quantities of a product have become or will become available for a client 12. For example, supply vector 662 may include information  
20 identifying a product from product list 660, a quantity of that product, a time at which the quantity will be or has become available, and the origin of the quantity of the product. When LFM 622 receives a component ATP request 32, LFM 622 uses supply vector 662 to determine if the requested quantity of the product is available and when. Using this and/or other information, LFM 622 may generate a component  
25 quotation 34 for that request 32.

Database 658 may also store accepted requests 664. Accepted requests 664 identify the component quotations 34, component promises 44, component acceptances 52, and/or component acceptance confirmations 54 that have been accepted by a client 12 and/or generated by LFM 622. The accepted requests 664  
30 could, for example, identify the client 12 to which a component promise 44 has been made, the product promised to the client 12, the quantity of the product that was promised to the client 12, the delivery date for the order, and a date when the

component promise 44 was made. Each accepted request 664 could be identified by an identifier such as a numeric code. A user, such as a supplier associated with LFM 662, may use the code, the date on which the component promises 44 were made, or any other suitable criteria to export accepted requests 664 from database 658 through data interface 656. In one embodiment, LFM 662 may provide a graphical user interface (GUI) or other interface to allow a user to identify which accepted requests 664 to export and to initiate exporting of the identified accepted requests 664.

Database 658 may also store peggings 666. A pegging 666 identifies a quantity of a product that is assigned or committed to a particular accepted request 664. For example, a pegging 666 may identify that ten units of a product to be produced next week have been committed to an order from a particular client 12. Multiple peggings 666 may be associated with one of the accepted requests 664. For example, another pegging 666 may identify that three hundred units of the same product to be produced this week have been committed to the order from the particular client 12. A pegging 666 may also identify the shipping date for the product, an identifier identifying the accepted request 664 associated with the pegging 666, and any other suitable information. A user may be allowed to export peggings 666 from database 658, for example, in a similar manner as for accepted requests 664 described above.

In addition, database 658 may store supply transactions 668 that affect the availability of a product. For example, a supply transaction 668 may identify the amount of a product that is currently in inventory due to delivery or production. Supply transactions 668 may also identify quantities of the product available due to returns of the product or cancellations of previous orders for the product. Supply transactions 668 could further include planned productions or procurements of the product. A user may be allowed to export supply transactions 668 from database 658, for example, in a similar manner as for accepted requests 664 described above. In one embodiment, a user may also be given the option of exporting supply transactions 668 or merely exporting the net available supply of a product, which factors in both supply and consumption.

Database 658 may store any other suitable information used by processor 650 to provide the order fulfillment functions of system 10. For example, database 658

could also store a hierarchy of sellers and/or a hierarchy of related products supported by fulfillment server 16. Other and/or additional information may be stored in database 658 without departing from the scope of the present invention.

Processor 650 may use the above and/or other information in any suitable  
5 combination to perform order fulfillment operations in system 10. For example, in one embodiment, LFM 622 may receive a component ATP request 32 from fulfillment server 16. Using product list 660, processor 650 may determine whether the product requested by the component ATP request 32 is available from the supplier or suppliers associated with LFM 622. If so, processor 650 may examine supply  
10 vector 622 and determine if and when the requested quantity of the product is or will be available. Processor 650 may generate a component quotation 34 using the information from database 658 and communicate the quotation 34 to fulfillment server 16.

Processor 650 may use any suitable method to determine whether the product  
15 requested by the component ATP request 32 is available from the supplier or suppliers associated with LFM 622. In one embodiment, processor 650 may use an "as soon as possible" approach to generating a component quotation 34. In a particular embodiment, the component ATP request 32 may include a parameter that represents a requested date and/or date range when the product is required to be  
20 shipped to the customer. Processor 650 may access supply vector 662 and begin searching for the requested quantity of the product in reverse chronological order starting on the requested ship date. Processor 650 continues searching backwards in time from the requested ship date until either the requested quantity is identified or the lower bound of the date range is reached. For each separate date on which the  
25 requested product is found to be available, processor 650 may add a shipment line-item to the quotation 34. If the lower bound is reached without finding a suitable quantity of the requested product, processor 650 may begin searching the supply vector 662 in chronological order starting on the requested ship date. Processor 650 continues searching forward in time from the requested ship date until either the  
30 requested quantity is identified or the upper bound of the date range is reached. If the total requested quantity of the product is found, processor 650 communicates the quotation 34 to fulfillment server 16. If less than the requested quantity of the product